kantox

Tomorrow's FX today

Milad Zaghab.

Kantox´s QA Job application.

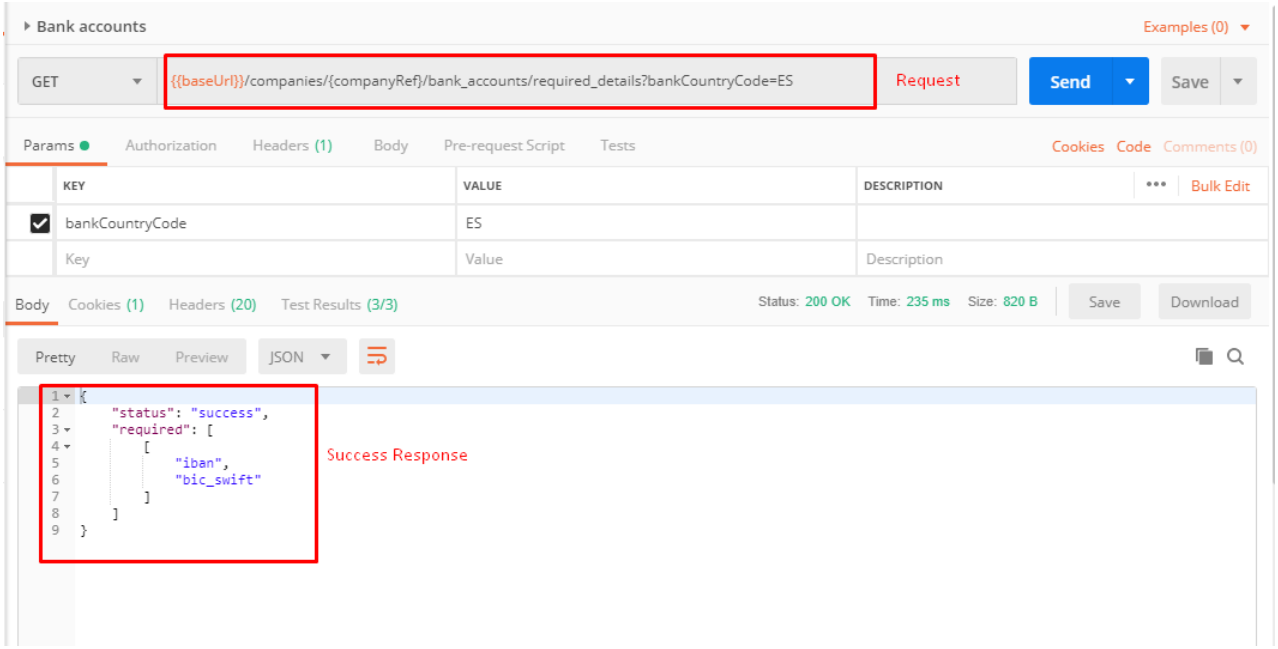
"*Quality means doing things well even when nobody is watching*"

Henry Ford.

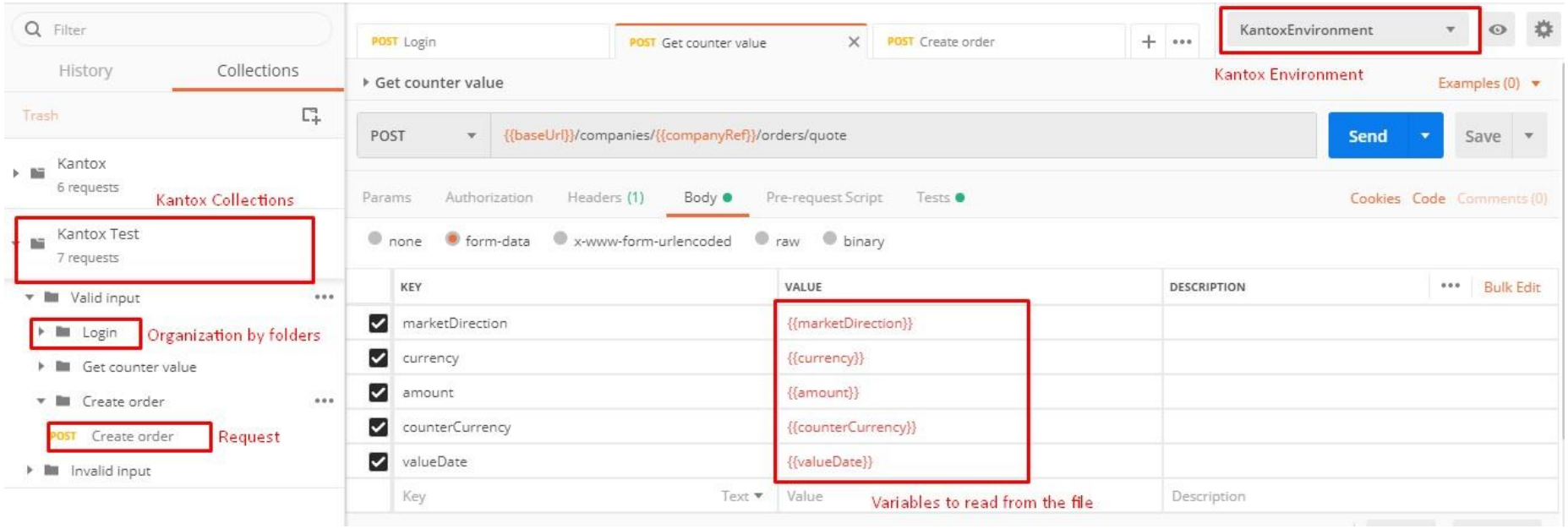
Kantox QA Excercise

The first part of the exercise, the exploratory session, was approached in two ways; the first one, was about getting to know the business rules and how the API was programmed, as well as looking at which details were required and what they mean in the context.

For the second approach, after studying some tools, I decided to work with Postman, it allowed me to get familiar with the API in an easy way. After studying the documentation provided by Kantox, a Test Case for the session was written, which can be find in the spreadsheet attached. To test it using Postman the exploratory test session took around 45 minutes, with the chosen tool it was possible to automate the token authorization process, to make some request and analyze the response, to organize and divide the work by environments and folders as well as providing unique data for each environment. Additionally, with the tool it was easy to change parameters, read and interpret the responses, evaluate the errors when a required data was not provided or when for example, wrong data type was provided.



Img 1. Bank Account request and response.



Img. 2 File organization and variable parametrization.

Test Case Automation

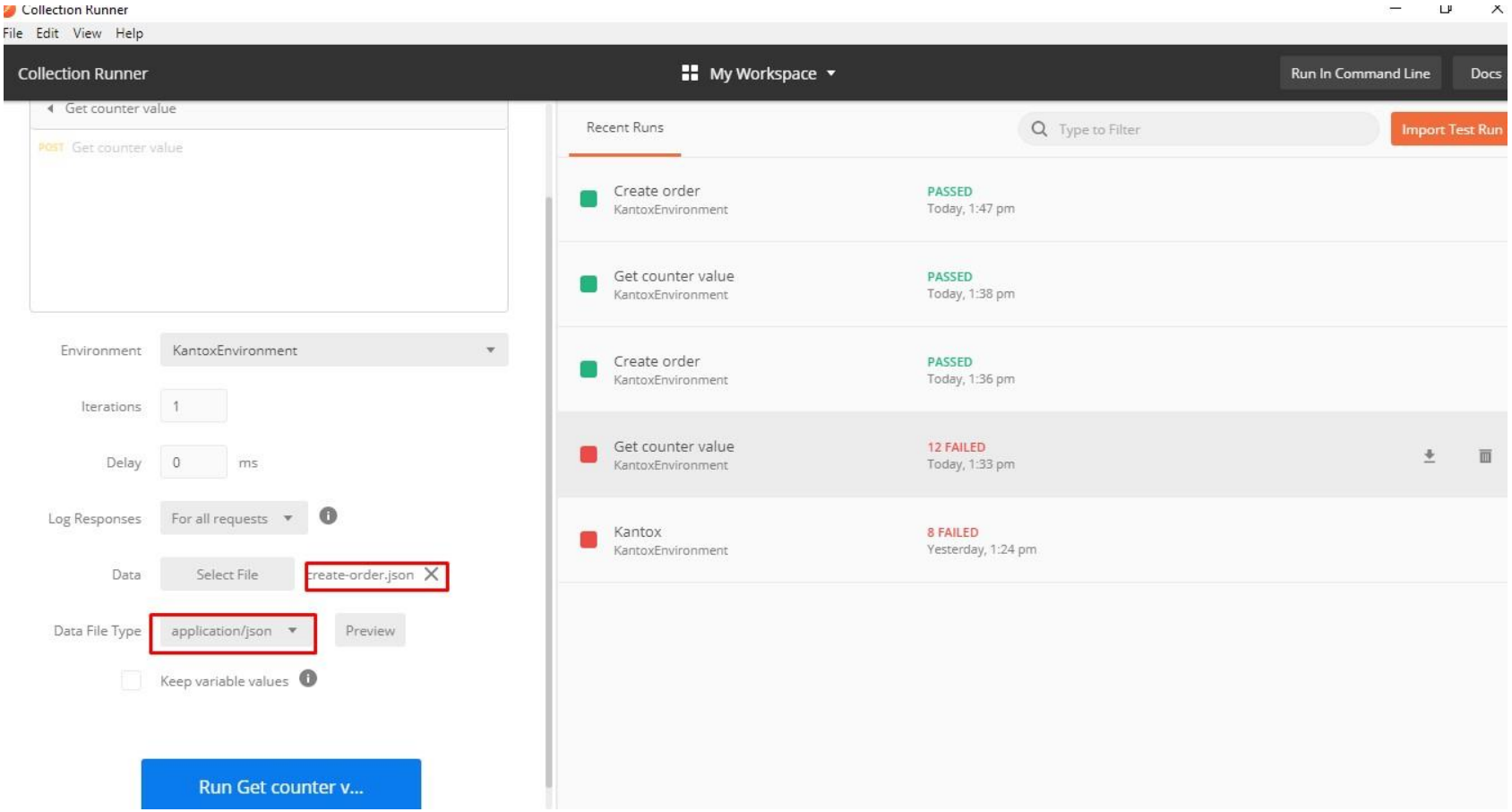
For the second section, test cases were written for the order´s management. The first one intention was to quote an order, the second one to create an order. The strategy used to test this section of the API was to create a Postman collection that allowed me to organize the requests in two folders, valid and invalid Data.

The parametrization for the requests was made with Postman collection variables, and from a data file. Collection variables were useful for testing common values used throughout requests, like the token, company reference, beneficiary account reference, login credentials, etc.

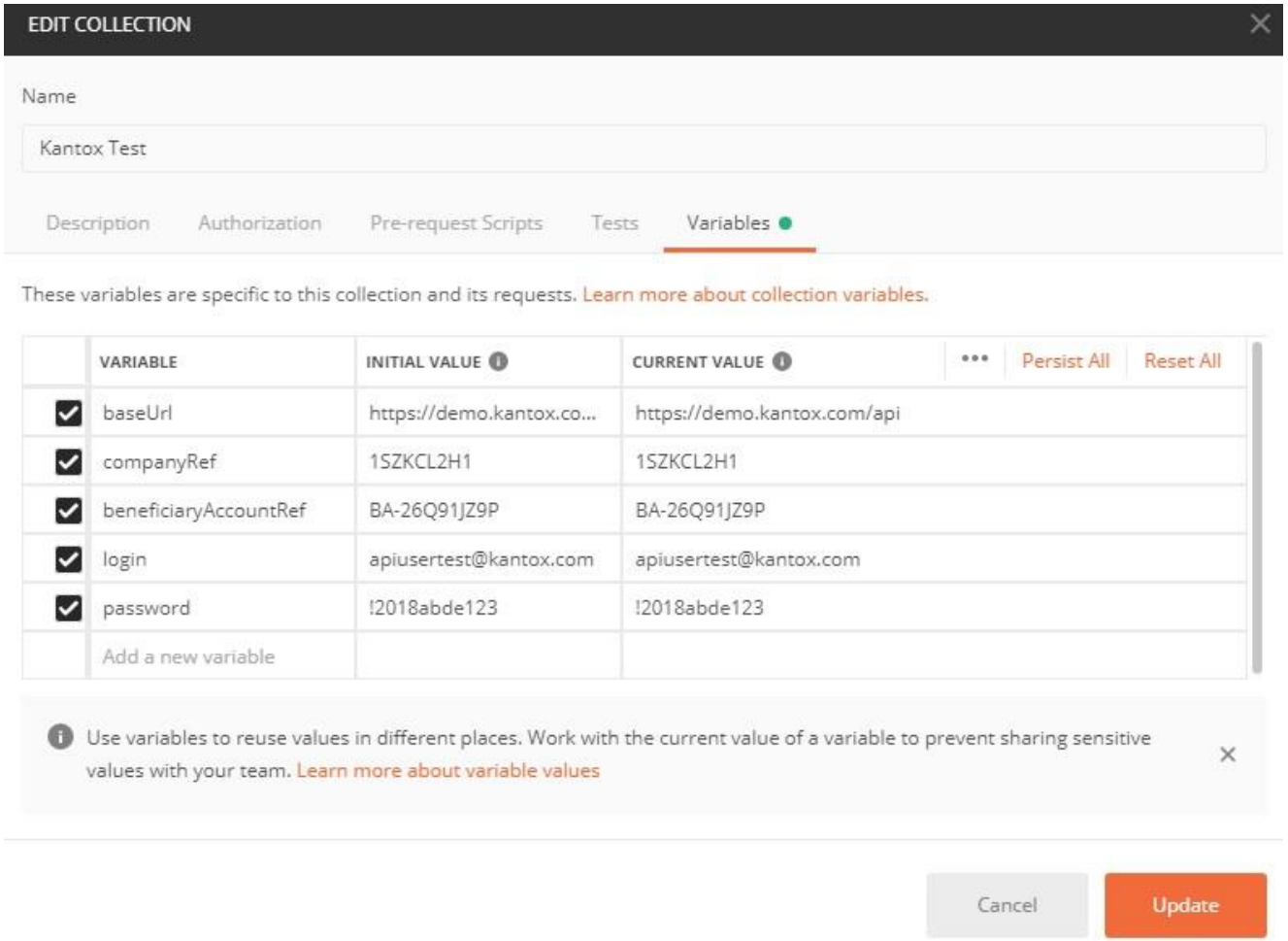
On the other hand, values from the data file offered a central, reusable and maintainable way to test the behavior of the endpoint on each request. It was used to assert the correct creation of the order with different values. To this end, multiple tests were written to check the response of the server. Things like the value of the status code, presence and non-emptiness of some fields were heavily tested to check that everything was ok.

Similarly, the behavior of the API with invalid data was also tested, checking that the appropriate error codes and fields were returned from the server.

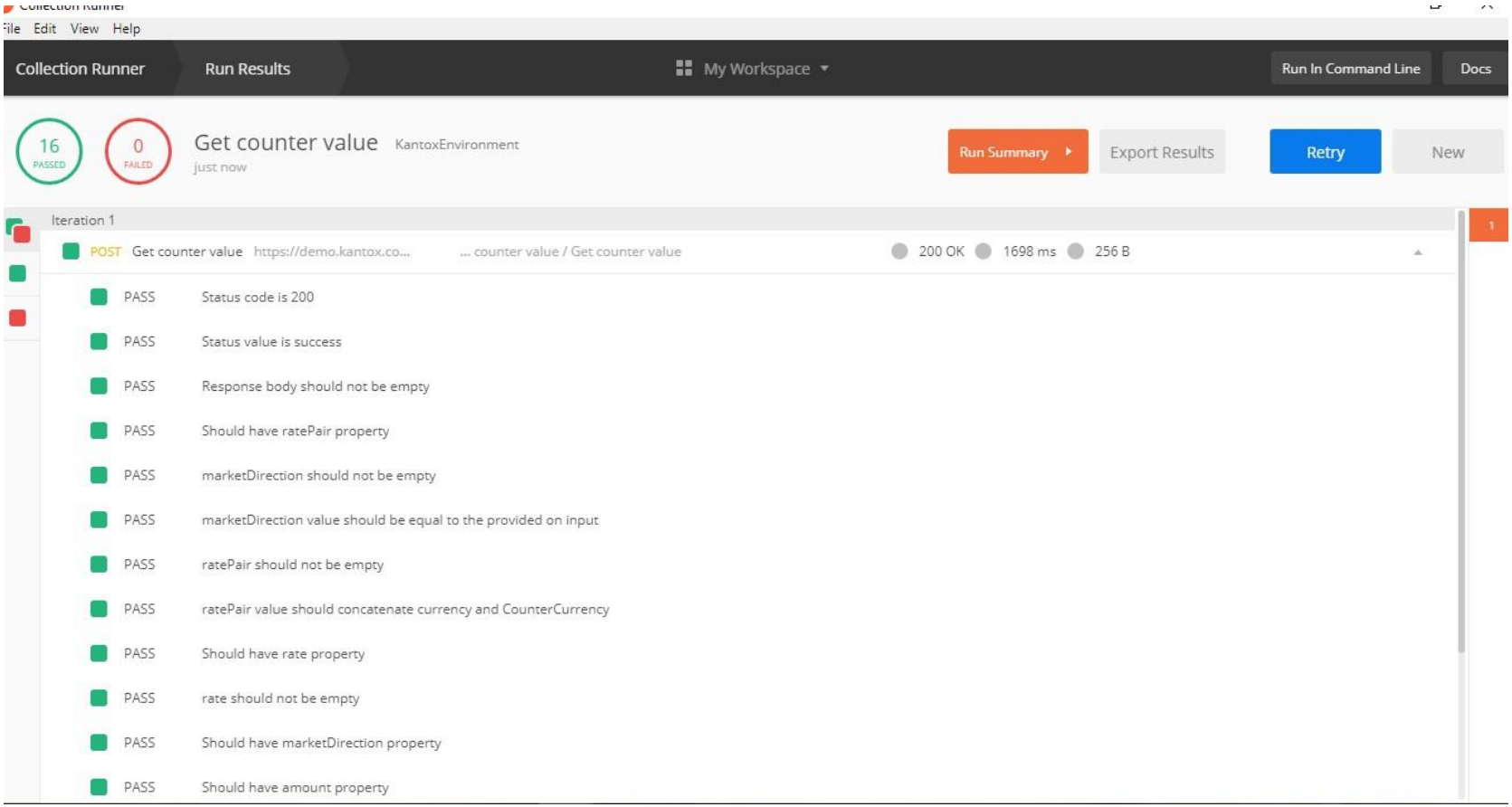
All in all, using Postman allowed a familiar, simple and maintainable way of testing the API and generating useful test runs summaries (with the collection runner); also, it allows to share the whole collection (with its variables) to another member of a team, so everyone can have the same base for testing.



Img. 3 Json file upload.



Img. 4 Collection´s variables.



Img. 5. Test results examples.